Adaptive Hypergraph Neural Network for Multi-Person Pose Estimation

Xixia Xu, Qi Zou*, Xue Lin

Beijing Key Laboratory of Traffic Data Analysis and Mining,
Beijing Jiaotong University, Beijing, China
{19112036,qzou,18112028}@bjtu.edu.cn

Abstract

This paper proposes a novel two-stage hypergraph-based framework, dubbed ADaptive Hypergraph Neural Network (AD-HNN) to estimate multiple human poses from a single image, with a keypoint localization network and an Adaptive-Pose Hypergraph Neural Network (AP-HNN) added onto the former network. For providing better guided representations of AP-HNN, we employ a Semantic Interaction Convolution (SIC) module within the initial localization network to acquire more explicit predictions. Build upon this, we design a novel adaptive hypergraph to represent a human body for capturing high-order semantic relations among different joints. Notably, it can adaptively adjust the relations between joints and seek the most reasonable structure for the variable poses to benefit the keypoint localization. These two stages are combined to be trained in an end-to-end fashion. Unlike traditional Graph Convolutional Networks (GCNs) that are based on a fixed tree structure, AP-HNN can deal with ambiguity in human pose estimation. Experimental results demonstrate that the AD-HNN achieves state-of-the-art performance both on the MS-COCO, MPII and CrowdPose datasets.

Introduction

Multi-person pose estimation (MPPE) aims to locate human keypoints for multiple persons in 2D image(Yang et al. 2017). It's basic to deal with many high-level tasks, such as action recognition(Yan et al. 2018) and 3D pose estimation(Li et al. 2020). Recently, although large progress has been made in pose estimation, while occlusions, variations in clothing, poses and viewpoints, unconstrained backgrounds remain challenge MPPE. It's a natural way to consider structural constraint among joints to handle these challenges. A serious of approaches employ a graph to model the relations between joints. Such graphs can mine human part relations and spatial context for improving MPPE performance. However, two properties of human poses are ignored to different extent in these methods: **(i)** high-order semantic dependencies among joints, **(ii)** relations of keypoints dynamically change subject to the variations in pose, viewpoint and occlusions should be considered.

There are some attempts (Wang et al. 2020; Qiu et al. 2020a; Jin et al. 2020) introducing graph convolutional network (GCN) to exploit the structural joints dependency for improving the 2D MPPE performance. Firstly, although these GCN-based methods have achieved promising results, they all treat the human body as a tree structure and represent it as a simplified graph as in Fig. 1(a). Namely, they only capture adjacent pair-wise joint relations and cannot model high-order semantic dependencies. Since the human body has a typical chain-like structure, the keypoint prediction is not only constrained by directly neighboring joints, but also is subject to multiple non-neighboring joints. This high-order joint relations can help infer the occluded keypoints (*i.e.*, like the challenging self-occlusions) with the global and local joint contexts. Such a complex relationship can hardly be captured by a simple graph with a set of fixed adjacent connections.

Secondly, the semantic relations between joints vary for different poses. It thus should build the most reasonable structure for dynamically changing poses. This makes it essential to automatically find the most suitable structure to benefit the estimation. A pre-defined initial graph that is built according to kinematic prior (like Fig. 1(**a**)) cannot provide such adaptability and robustness. Despite the DGCN(Qiu et al. 2020b) explores the high-order relations with a dynamic GCN, while it assumes the location subjects to a dis-

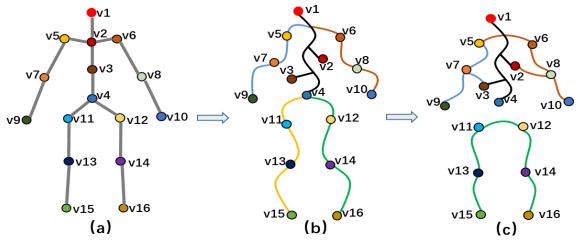

Figure 1: Graph-based human body representation, (a) a simple graph (tree structure), (b) an initial hypergraph with adaptability, (c) an adjusted hypergraph. For edges in (a), both the edge degree and the number of edges are fixed. By comparison, the hyperedges are free in (b). As the model trains, new (old) hyperedges will create (disappear) and assignments of joints belonging to a hyperedge will change. Finally, the hypergraph structure can be learned as (c). Different colored lines indicate different IDs of hyperedges.

*corresponding author.

tribution and cannot adjust the structure adaptively.

Aiming at the above issues, we creatively propose to represent a human body as an adaptive hypergraph(Feng et al. 2019) as depicted in Fig. 1**(b)** aiming to learn local and global structural dependencies among joints flexibly. Unlike a simple graph, an adaptive hypergraph represents the kinematic connections with flexible hyperedges. The hyperedges have no fixed degrees and connect different joints freely according to different semantic relations in different poses. This characteristic enables it capture the high-order structural dependencies among joints adaptively and learns an optimal structure as the model trains as in Fig. 1**(c)**.

Upon the above representations, this paper develops a novel **AD**aptive **H**ypergraph **N**eural **N**etwork **(AD-HNN)** for 2D MPPE as in Fig. 2. It mainly comprises a keypoint localization network and an **A**daptive-**P**ose **H**ypergraph **N**eural **N**etwork **(AP-HNN)**. The AP-HNN is added onto the former network and constructed with the heatmap predictions. For providing better initialization, we design a **S**emantic **I**nteraction **C**onvolution **(SIC)** module within stage-1 to regress more precise predictions via exploring the feature relationships. Build on this, we construct an AP-HNN to flexibly capture high-order joint dependency among joints. Instead of adopting a fixed structure, the most reasonable pose structure can be discovered by seeking the potential joint semantic relations. The two-stages are combined together to be trained in an end-to-end manner.

Overall, our main contributions are as follows:

• We develop an adaptive hypergraph-based two-stage MPPE method, AD-HNN, which brings significant improvement. A SIC module is proposed in initial stage to explore the feature relations for more accurate predictions.

• A novel adaptive hypergraph is designed to flexibly consider the reasonable connections among joints for variable poses and, hence, to capture the high-order joint relations. To our knowledge, this is the first work that brings the advantages of hypergraph to 2D MPPE.

• The proposed AD-HNN is extensively validated on MS-COCO, MPII and CrowdPose datasets and achieves an outstanding performance.

Related Work

Multi-Person Pose Estimation. Recently, researchers have made much efforts(Cao et al. 2019; Luvizon and Picard 2019; Wei et al. 2016; Sun et al. 2017; Tang, Yu, and Wu 2018; Ning, Zhang, and He 2018; Chu et al. 2017; Chou, Chien, and Chen 2018; Chen et al. 2017; Ke et al. 2018; Huang et al. 2020; Moon, Chang, and Lee 2019; Zhang et al. 2020) on HPE to accelerate its progress. Two mainstream methods are prevalent in MPPE including bottom-up(Insafutdinov et al. 2016; Newell and Deng 2017; Papandreou et al. 2017; Cheng et al. 2020) and top-down(Newell and Kaiyu 2016; Fang et al. 2017; Chen et al. 2018; Xiao and Wei 2018; Sun et al. 2019; Su et al. 2019). This paper follows the top-down pipeline. The recent graph-based methods(Jin et al. 2020; Qiu et al. 2020b; Wang et al. 2020) model the joint relations and achieves great performance. In light of this, we model the human body with a hypergraph which is a general form of the simple graph.

Graph-based MPPE. The graph representation for 2D MPPE is not new. Recently, DGCN(Qiu et al. 2020b) construct dynamic graph to tolerate large pose variations. OPEC(Qiu et al. 2020a) design an Image-Guided Progressive GCN to estimate the invisible joints. Gpcnn(Wang et al. 2020) embeds a graph pose refinement module to model the human structure. HGG(Jin et al. 2020) propose a differentiable graph grouping method to assign keypoints. However, all above GNN-based methods are based on a simple graph which only considers the one-order structural joint relations.

Hypergraph Learning. Recently, HNN(Feng et al. 2019) learns multi-modal data relations by hyperedge convolution. Unlike an edge in common graphs only connects two vertexes, a hyperedge connects two or more vertexes. It satisfies the characteristics of high-order structural relations in human keypoints. Some works(Yu and Tao 2012; Zhu et al. 2017) adopt the hypergraph to learn high-order data relations. They treat each sample as one vertex and optimize model by fixing others. DHNN(Jiang et al. 2019) constructs the hypergraph dynamically due to the limitation of the fixed structure. Inspired by this, we extend GCN to HNN and design an adaptive learning mechanism to capture the high-order human kinematics flexibly.

There are researches introducing the hypergraph to study the human-related computer vision task. For example, the (Kim et al. 2020) design a hypergraph attention network to define semantic modality relations and combine multi-modal features. The hyperReID(Yan et al. 2020) propose a multi-granular hypergraph to model the multi-granular spatio-temporal dependency. The most related work is the SD-HNN(Liu et al. 2020), which adopts hypergraph to represent the human body exploiting the joint relations for 3D pose estimation. However, it designs a semi-dynamic graph by introducing an extra adaptive matrix, this makes it hard to train. Distinctly, we propose a flexible adaptive learning mechanism to stably adjust the hypergraph structure.

Method

Semantic Interaction Convolution

Given the similar visual patterns of joints, the semantic representations of feature maps are highly correlated in spatial distributions. We introduce the SIC module to model the correlation among feature channels, where each channel encodes a visual pattern related to specific keypoint. Intuitively, the feature channels with similar semantics would be activated simultaneously when a specific pattern of keypoints emerges. By grouping feature maps with similar semantics, it finds the latent keypoint visual patterns to improve the performance. The initial pose estimation model is built on SBN(Xiao and Wei 2018), the SIC followed by different deconvolutional layers to explore the feature relations from multi-resolution feature maps as in Fig. 3.

Feature Interaction Learning. Measured by the feature distance in a space, the neighboring feature maps with similar patterns form a meaningful subset. The feature map set $F = \{fe_1, fe_2, ..., fe_c\}$ consists of the deconvolutional features. For each fe_i, we found its k-nearest neighbors ($k = 5$). The relation between a pair of neighboring feature

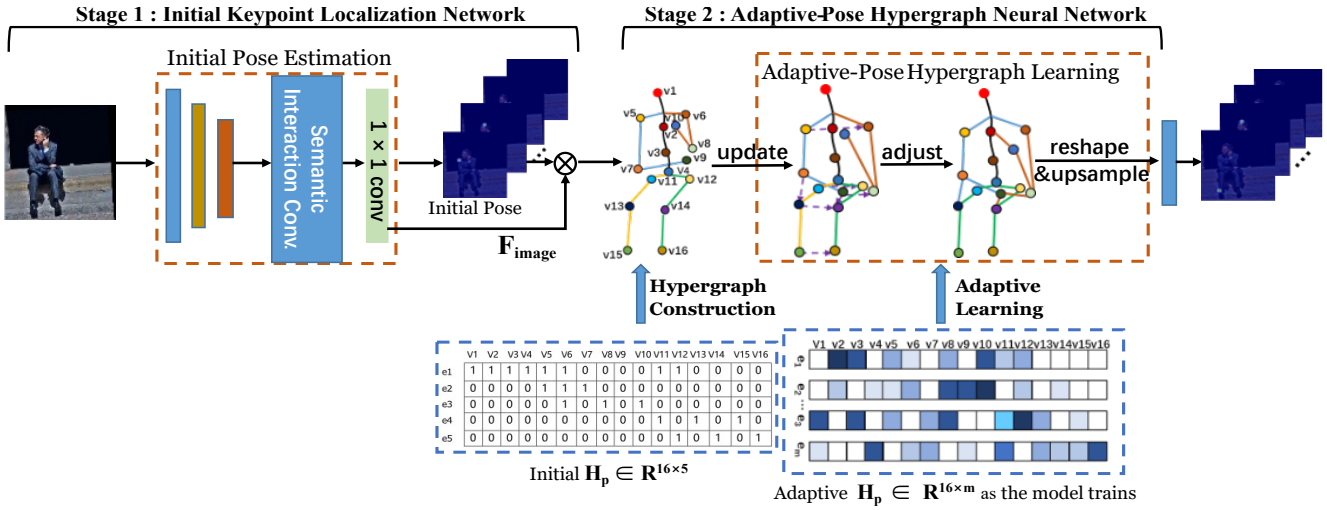

Figure 2: Overview of the proposed AD-HNN. In stage-1, an initial pose estimator equipped with our SIC module obtain the initial heatmap predictions. In stage-2, an adaptive hypergraph builds a human body to adaptively correlate the adjacent or non-adjacent joints and explore the high-order joint semantics. Each colored line depicts a hyperedge. The adjusted hyperedge in learned structure connects joints with potential relations that is more consistent with the human commonsense.

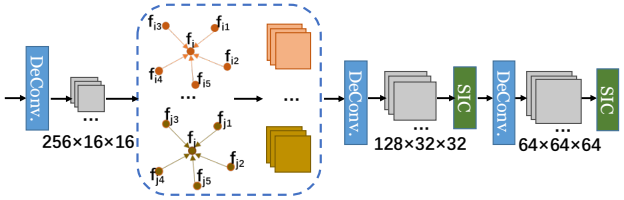

Figure 3: The process of Semantic Interaction Convolution.

maps is as below:

$$e'_{ij} = ReLu(fe_j - fe_i), fe_j \in \mathcal{N}_{fe_i}, \tag{1}$$

where $\mathcal{N}_{fe_i}$ is the neighbors of fe_i. For each fe_i, we consider its closest k feature maps and compute their cosine distance. Finally, a channel-wise max function operates on e'_{ij} to capture the discriminative features. The SIC output for the i-th feature:

$$fe'_i = \max_{j:fe_j \in \mathcal{N}_{fe_i}} e'_{ij}. \tag{2}$$

Mapping on Keypoint Predictions. Generally, the joint predictions benefit from the relation modeling ability of SIC and the spatial representation of heatmaps. Each joint location is associated with the encoded feature channel, which can be influenced by the learned feature combinations from SIC. The learned feature set can explain the specific visual patterns of keypoints via capturing the local and inherent feature relations. The learned weights describe the association between the features with the keypoint predictions, and a larger one leads to more explicit joint location in the pattern. In this way, the joint predictions can be more explicit by activating different visual patterns.

Adaptive Hypergraph for MPPE

Review of GCN and HNN. Assume that a graph is $G = \{A, X\}$. The adjacency matrix is $A = \{a_{ij}\} \in R^{n\times n}$ that depicting the node connections, while $a_{ij} > 0$ means there exists an edge between node i and j. The node set is $X \in R^{n\times d}$, n is the vertex numbers, d is the feature dimensions. Based on above terminologies, a convolution of GCN(Kipf and Welling 2016) can be depicted as:

$$H^{(l+1)} = \sigma(\tilde{A}H^{(l)}W^{(l+1)}), \tag{3}$$

where $H^{(l)}$ denotes the node representations in l-th layer, W depicts the parameters, $\sigma(\cdot)$ is a non-linear activation function. $\tilde{A}$ is the normalized adjacency matrix.

The concept of GCN is extended to hypergraph in(Feng et al. 2019) and a new hypergraph neural network is proposed. The convolution of HNN is formulated as:

$$X^{(l+1)} = \sigma(D_v^{-\frac{1}{2}}HWD_e^{-1}H^TD_v^{-\frac{1}{2}}X^{(l)}\Gamma^{(l)}), \tag{4}$$

where D_v, D_e denote the diagonal matrices of the vertex degrees and edge degrees respectively. The H denotes the incidence matrix of hypergraph, W depicts the diagonal matrix of the hyperedge weights and the filter Γ is applied over the hypergraph nodes to extract features.

Motivation. In GNN-based works, the human body is often represented as a fixed tree structure, where the connections among adjacent joints only represent adjacent pair-wise relations, but not the high-order semantic relations among joints. The pairwise relationship cannot adjust to potential non-physical connections flexibly. For example, for the pose of stretching legs like in Fig. 4, the 'rwrist' and 'rankle' have a close connection but this relations cannot be captured by GNN. Intuitively, we involve a novel hypergraph to describe this flexible and complex kinematics. The characteristics of hypergraph make it capture more comprehensive human context to help correct the inaccurate predictions.

Commonly, the hypergraph structure is fixed according the predefined physical connections. However, the learned

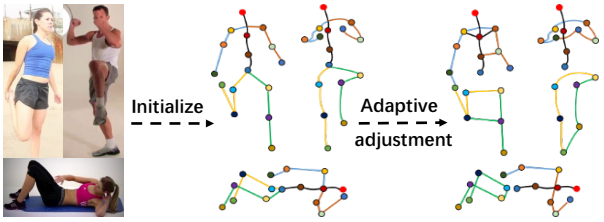

Figure 4: The examples of the adaptive adjustment process.

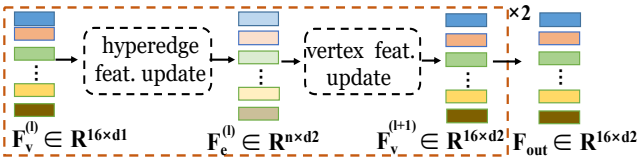

Figure 5: The illustration of the adaptive hypergraph convolution learning. '×2' denotes that two layers are adopted.

structures of hypergraphs are not always same for different poses. For example, the number of hyperedges of pose 'leg stretch' are 5 but for 'crunches' are 4 because all the symmetrical lower body parts have similar semantic relations and they finally are connected by a same hyperedge. However, this flexible relations cannot be described by the fixed structure. We should make the hypergraph adjust its structure to fit for the flexible joint relations adaptively. To achieve this, we adjust the hypergraph structure to connect those joints which may have semantic relations into same hyperedge via learning the incidence matrix as the model trains. In this manner, the hypergraph achieves to capture high-order joint relations flexibly indeed. It maximizes the learning ability of pose hypergraph and thus can deal with the flexible pose variations.

Hypergraph Construction. Following (Feng et al. 2019), our adaptive hypergraph is defined as $\mathcal{G} = (\mathcal{V}, \xi, W_e)$, where $\mathcal{V} = \{v_1, v_2, ..., v_n\}$ denotes the vertex set including n joints, $\xi = \{e_1, e_2, ..., e_m\}$ denotes the hyperedge set and W_e is a diagonal matrix of hyperedge weights, which initialized with an identity matrix of meaning equal weights. The hypergraph incidence matrix is denoted as $H_p \in \mathbb{R}^{n\times m}$. We redefine a hyperedge indicating a potential relation between two or more joints, rather than a specific prior relationship.

As shown in Fig. 2, given an input, processed by the stage-1, we get the initial heatmap predictions $\hat{P}_{heat}$. Then, we acquire the vertex features F_v to form the initial hypergraph.

$$F_v = GAP(F_{image} \odot \hat{P}^i_{heat}), i \in 1, 2..n, \tag{5}$$

where F_{image} denotes the visual features, $F_v \in \mathbb{R}^{n\times d_1} = \{f_1, f_2, .., f_n\}$ as shown in Fig. 5, the d_1 denotes the feature dimension. Initially, hyperedges are set as 5 by the predefined kinematic chains, each hyperedge corresponding to the kinematic chain connects all the joints in that chain. The $H_p \in \mathbb{R}^{n\times 5}$ acts as the basic structure as in shown Fig. 2.

Adaptive Hypergraph. We develop an adaptive hypergraph via updating the H_p adaptively during training. For learning the relevance between the joints and hyperedges, we apply an adaptive mechanism like attention on H_p to present a probabilistic model, which assigns non-binary and real values to measure the degree of connectivity. For a given vertex v_i, we compute its relevance with the other hyperedges e_m,

$$H_p = ave(\sigma(sim(f_i\Phi, f_j\Phi))), v_j \in e_m, \tag{6}$$

$$\tilde{H}_p = \frac{exp(H_p)}{\sum_{e_m\in\xi} exp(H_p)}, \tag{7}$$

where $\Phi \in \mathbb{R}^{d_i\times d_j}$ is the learned weighting matrix, the v_j belongs to e_m, $sim(\cdot)$ computes a cosine similarity between vertex features.

Concretely, if the relevance between the specific vertex and hyperedge is lower than a specific score (0.75 used here), they shouldn't be built connections. And if there exists two hyperedges with stronger relevance, we merge them as a same hyperedge. After the first round of learning, we update H_p as in Eq. 6, 7, and adjust the hypergraph structure for further learning to adapt more reasonable relations. As shown in Fig. 2, the H_p adaptively updates as the model learns, which enables the initial hypergraph structure adjust to the most reasonable structure.

High-order Semantic Learning. To capture high-order semantic joint relations, the two-layer hypergraph convolution with adaptive hypergraph is adopted. Let $\Phi^{(l)}$ denotes the learnable filter matrix of the $\mathcal{G}$ at l-th layer, $F^{(l)}$ is the vertex features and $F(0) = F_v$. A hypergraph convolutional layer(Feng et al. 2019) can be depicted as:

$$F^{(l+1)} = \sigma(D_v^{-\frac{1}{2}} H_p W_e D_e^{-1} H_p^T D_v^{-\frac{1}{2}} F^{(l)} \Phi^{(l)}), \tag{8}$$

where $\sigma(\cdot)$ is the activation function, D_e and D_v are used for normalization. For a vertex $v \in \mathcal{V}$ and a hyperedge $e \in \xi$, their degrees can be computed by

$$d(v) = \sum_{e\in\xi} w(e)a(v, e), d(e) = \sum_{v\in\mathcal{V}} a(v, e), \tag{9}$$

where $w(e)$ is the weight of e, $a(v, e)$ is the element of H_p.

As the Eq. 8 and Fig. 5 depicted, we adopt the learnable matrix $\Phi^{(l)}$ to transform the vertex features $F_v^{(l)}$ to the new vertex features. Then, the updated vertex features on the hyperedges are gathered to obtain the hyperedge features by multiplying H_p^T. Finally, the related hyperedge features are associated to obtain the final vertex features $F_v^{(l+1)}$, which is achieved by multiplying the H_p. Through the vertex-hyperedge-vertex transform, AP-HNN can adaptively capture the joint dependencies and explore their high-order semantic interactions.

Optimization

We denote the ground truth pose over training set Ω as P_i, and the output as $\hat{P}_i$. Following the SIC, we produce an initial heatmap-based prediction $\hat{P}_{heat}$. Hence, the total loss is the weighted sum of the initial heatmap-based loss and the hypergraph-based loss of final prediction:

$$\mathcal{L}_{total} = \min_{\theta} \sum_{i\in\Omega} (\lambda_1 \mathcal{L}(\hat{P}_{heat}, P_i) + \lambda_2 \mathcal{L}(\hat{P}_i, P_i)), \tag{10}$$

where θ denotes all the trainable parameters of the model and λ_1 and λ_2 are 0.3, 0.7 respectively.

Experiments

Datasets and Evaluation Metric

COCO Keypoint Detection(Lin et al. 2014) includes about $57K$ images for training and $5K$ images for validation, $20K$ images for testing. The evaluation metrics adopt OKS-based average precision (AP) and average recall (AR).

MPII Human Pose Dataset includes about $25K$ images with $40K$ objects, where there are $12K$ objects for testing and the remaining for training. We use the standard metric PCKh(Andriluka et al. 2014) (head-normalized probability of correct keypoint) score as evaluation.

CrowdPose(Li et al. 2019) contains $20K$ images and $80K$ human instances, which aims to promote performance in crowded cases and uses the same evaluation with COCO. It divides into three crowding levels by Crowd Index: *Easy* ($0 \sim 0.1$), *Medium* ($0.1 \sim 0.8$) and *Hard* ($0.8 \sim 1$).

Implementation Details

Network Architectures. We adopt HRNet(Sun et al. 2019) and SBN(Xiao and Wei 2018) as the initial pose estimator and the backbone adopts ResNet-152 and HRNet-w32 in default. It's noted we only add AP-HNN on HRNet without SIC. We adopt two HyperConv layers followed by BN and RELU and repeat four times.

Training. We implement all experiments in PyTorch on a single NVIDIA TITAN XP GPU. Our models are initialized with the weights of the pretrained ImageNet(Russakovsky et al. 2015). For MS-COCO, human detection boxes are resized to 384×288. The Adam(Kingma and Ba 2015) adopts the learning rate with 10^{-3} and reduced to 10^{-4} and 10^{-5} at $170th$ and $200th$ epochs. For MPII, input size is 384×384 and trained for 180 epochs. For CrowdPose, the training setting is similar with COCO and trained for 220 epochs. For augmentation, we follow the HRNet(Sun et al. 2019).

Quantitative Results

Comparison on MPII. We evaluate the PCKh@0.5 on MPII in Tab. 1. Compared with the existing methods, we achieve the best performance with the same backbone. For example, we achieve 92.0% score with ResNet, and boosts 0.5% than the baseline (SBN). The trend also fits for HRNet-based model. And our model largely lifts 11.1% score over the graph-based DGCN(Qiu et al. 2020b). This shows that adopting hypergraph to model human pose has greater advantage than the common graph.

Comparison on CrowdPose. To show our method is robust in crowd scenes, we conduct experiments on CrowdPose in Tab. 2. Our AD-HNN lifts 0.2 mAP over the best GCN-based OPEC(Qiu et al. 2020a) that is specially designed for occlusions, and outperforms others on all metrics except AP_{50}. We also list AP at different crowded levels. Improvements remain high even at the high crowd level.

This proves that our adaptive hypergraph can better deal with the occlusions or pose variations than the common GCN due to its adaptability to deal with flexible variations and richer context representation to infer occluded joints.

Method	Hea	Sho	Elb	Wri	Hip	Kne	Ank	Total
Wei(2016)	97.8	95.0	88.7	84.0	88.4	82.8	79.4	88.5
Newell(2016)	98.2	96.3	91.2	87.2	89.8	87.4	83.6	90.9
Sun(2017)	98.1	96.2	91.2	87.2	89.8	87.4	84.1	91.0
Ning(2018)	98.1	96.3	92.2	87.8	90.6	87.6	82.7	91.2
Luvizon(2017)	98.1	96.6	92.0	87.5	90.6	88.0	82.7	91.2
Chu(2017)	98.5	96.3	91.9	88.1	90.6	88.0	85.0	91.5
Chou(2018)	98.2	96.8	92.2	88.0	91.3	89.1	84.9	91.8
Chen(2017)	98.1	96.5	92.5	88.5	90.2	89.6	86.0	91.9
Yang(2017)	98.5	96.7	92.5	88.7	91.1	88.6	86.0	92.0
Ke(2018)	98.5	96.8	92.7	88.4	90.6	89.3	86.3	92.1
Tang(2018)	98.4	96.9	92.6	88.7	91.8	89.4	86.2	92.3
DGCN(2020)	95.6	92.5	83.1	76.5	81.5	73.1	65.1	81.2
SBN(2018)	98.5	96.6	91.9	87.6	91.1	88.1	84.1	91.5
HRNet(2019)	98.6	96.9	92.8	89.0	91.5	89.0	85.7	92.3
Ours (ResNet)	98.5	96.6	92.6	88.5	91.0	88.8	86.0	92.0
Ours (HRNet)	98.6	96.9	92.8	89.2	91.6	89.3	86.3	**92.4**

Table 1: Result comparisons on the MPII *test set*.

Method	AP	AP_{50}	AP_{75}	AP_E	AP_M	AP_H
OpenPose(2016)	-	-	-	62.7	48.7	32.3
Mask-RCNN(2017)	57.2	83.5	60.3	69.4	57.9	45.8
SBN(2018)	60.8	81.4	65.7	71.4	61.2	51.2
RMPE(2017)	61.0	81.3	66.0	71.2	61.4	51.1
HigherHRNet(2020)	65.9	86.4	70.6	73.3	66.5	57.9
CrowdPose(2019)	66.0	84.2	71.5	75.5	66.3	57.4
OPEC(2020)	70.6	**86.8**	75.6	-	-	-
Ours	**70.8**	86.5	**75.8**	**77.0**	**68.5**	**59.6**

Table 2: Performance comparisons on CrowdPose *test set*.

Comparison on MS-COCO. The AD-HNN achieves the best result at most of metrics and surpasses other methods largely on COCO *test-dev* in Tab. 3. Notably, AD-HNN exceeds the previous best result (HRNet: 75.5%) by 1.1%.

Compared with most of the recent graph-based methods, *e.g.*, OPEC(Qiu et al. 2020a), Dgcn(Qiu et al. 2020b) and HGG(Jin et al. 2020), AD-HNN also achieves a notable improvement on all metrics, which demonstrates its effectiveness. Specifically, it has 1.2% higher than the OPEC and achieves a comparable result with the best Gpcnn(Wang et al. 2020). It proves the superiority of an adaptive hypergraph capturing semantic high-order relations among joints over a fixed graph. Notably, although Gpcnn achieves the best result, the guided point sampling and the supervision from hard negative (positive) samples make a great contribution in performance except the graph model.

For illustrating the superiority of AP-HNN for the extreme self-occlusions, we choose 185 images with 570 samples from the MSCOCO *val2017*. Tab. 4 shows our method improves 3.5% mAP over the baseline. This indicates our adaptive hypergraph helps to infer the occluded joints by the

Method	Back	size	AP	AP_{50}	AP_{75}	AP_M	AP_L	AR
RMI(2017)	r101	353×257	64.9	85.5	71.3	62.3	70.0	69.7
AE(2017)	-	512×512	65.5	86.8	72.3	60.6	72.6	70.2
CPN(2018)	Ince	384×288	73.0	91.7	80.9	69.5	78.1	79.0
CSA(2019)	r152	384×288	74.5	91.7	82.1	71.2	80.2	80.7
PFix(2019)	r152	384×288	**76.7**	92.6	**84.1**	**73.1**	**82.6**	81.5
Gpcnn(2020)	r152	384×288	**75.1**	91.8	**82.3**	**71.6**	**81.4**	80.2
OPEC(2020)	-	384×288	73.9	**91.9**	82.2	-	-	-
Dgcn(2020)	r152	641×641	67.4	88.0	74.4	63.6	73.0	73.2
HGG(2020)	4HG	512×512	67.6	85.1	73.7	62.7	74.6	71.3
SBN(2018)	r152	384×288	73.7	**91.9**	81.1	70.3	80.0	79.0
UDP(2020)	r152	384×288	74.7	91.8	82.1	71.5	80.8	80.0
Ours	r152	384×288	**75.1**	91.7	**82.5**	**71.5**	**81.3**	80.1
HRNet(2019)	hr32	384×288	74.9	92.5	82.8	71.3	80.9	80.1
UDP(2020)	hr32	384×288	76.1	92.5	83.5	72.8	82.0	81.3
Gpcnn(2020)	hr32	384×288	**76.4**	**92.5**	**83.8**	**72.9**	**82.4**	80.3
Ours	hr32	384×288	76.3	92.3	83.8	73.0	82.2	81.2
HRNet(2019)	hr48	384×288	75.5	92.5	83.3	71.9	81.5	80.5
UDP(2020)	hr48	384×288	76.5	**92.7**	84.0	73.0	82.4	81.6
Dark(2020)	hr48	384×288	76.2	92.5	83.6	72.5	82.4	81.1
Ours	hr48	384×288	**76.6**	92.4	**84.3**	**73.2**	**82.5**	81.5

Table 3: Performance comparisons with the state-of-the-arts on COCO *test-dev2017*. The first row denotes the classical methods and second row depicts GNN-based methods.

aid of the learned high-order joint context. For the results on the self-occlusion subset are better than results on the whole set possibly because there exists normal postures in a self-occluded image but we build this on the image-level.

Ablation Study

Effect of the Components. We compare with the baseline (ResNet-152) to illustrate the effectiveness of the proposed modules in Tab. 4. Our AP-HNN and SIC module have 2.1%, 0.5% improvements than the baseline respectively. The adaptive hypergraph model really makes a big contribution by establishing more reasonable connections among different joints flexibly. As well, the semantic interaction convolution module further improves the accuracy.

Methods	AP	AP_{50}	AP_{75}	AP_M	AP_L	AR
Baseline	74.3	89.6	81.1	70.5	79.7	79.7
w/ SIC	74.8	89.8	81.5	70.9	80.2	80.0
w/AP-HNN	75.9	90.6	82.7	71.9	82.3	81.0
Ours(ensemble)	**76.4**	**90.8**	**82.8**	**72.3**	**83.3**	**81.2**
self-occlusion subset						
Baseline(**val-sub**)	76.0	94.5	84.4	71.2	83.2	79.3
Ours(**val-sub**)	**79.5**	**95.8**	**86.9**	**74.2**	**85.8**	**81.4**

Table 4: The effect of the proposed components on *val2017*.

HNN vs. GNN. To explore the effect of the HNN, we replace the AP-HNN by the GNN and HNN, respectively in Tab. 5. The graph construction adopts the fixed tree-like structure. The GNN-based follows OPEC(Qiu et al. 2020a). The result drops from 75.3% to 74.9% when replace HNN with GNN. This proves that HNN is superior to GNN in exploring the high-order structural dependency.

Methods	AP	AP_{50}	AP_{75}	AP_M	AP_L	AR
Baseline	74.3	89.6	81.1	70.5	79.7	79.7
GNN	74.9	90.1	81.8	70.9	80.3	80.2
Static HNN	75.3	90.4	82.2	71.4	81.2	80.6
Dynamic HNN	75.6	90.4	82.3	71.6	81.8	80.8
Adaptive HNN	**75.9**	**90.6**	**82.7**	**71.9**	**82.3**	**81.0**

Table 5: Comparisons of different hypergraph learning strategies on *val2017*.

Effect of Adaptive Hypergraph. To evaluate the effectiveness of our adaptive hypergraph, we compare different HNN-based methods in Tab. 5. We **firstly** explore the difference between Dynamic HNN and Static HNN(joint relations fixed). The Dynamic HNN(Jiang et al. 2019) builds the keypoint connections dynamically which updates the features via k-NN algorithm. The mAP ups to 75.6% from 75.3%, indicating the advantage of dynamic HNN in adapting to changes of human structures over fixed ones. Still, the parameters of the incidence matrix of static HNN are frozen and not learnable during training. **Secondly**, we compare adaptive HNN with Dynamic HNN. For Dynamic HNN, it's hard to find out the most reasonable joint relations because it adjusts the structure solely relying on the vertex features rather than considering the flexible semantic relevance between the vertex and other hyperedges. Besides, the hyperedge numbers of Dynamic HNN are fixed. In contrast, our Adaptive HNN can automatically build the most reasonable joint relations in a learnable way. The result also proves the effectiveness of the adaptive learning for pose variations.

The Generality of AP-HNN on HPE. We validate the AP-HNN on different 1-stage methods in Tab. 6. We adopt four top-down and bottom-up methods respectively to provide the initialization. The result shows consistent improvements owing to the AP-HNN. It proves that AP-HNN can better help correct the initial inaccurate predictions benefiting from the learned high-order structure information and its adaptability for complex postures. Also, AP-HNN has good generality to different heatmap-based methods.

Visualization and Analysis

Component Analysis. Fig. 6 visualizes the results of different modules on AD-HNN. Columns 2, 3, and 4 depict the results of using Convolution (Conv.) instead of SIC, fixed HNN instead of Adaptive Hypergraph (Adap.), and using GNN to replace HNN, respectively. Compared with less accurate locations and relatively low confident keypoints got by GNN, HNN, and Conv., the keypoint-aware areas of AD-HNN are more concentrated. It demonstrates the effectiveness of the proposed components.

Comparison Analysis. Noted that we don't compare with most of the graph-based methods in qualitative since their codes are not released. To show the advantage of AP-HNN, we replace it with the GNN model onto our initial models.

1-Stage Method	AP	AP_{50}	AP_{75}	AP_M	AP_L	AR
w/o AP-HNN						
Bottom-up						
OpenPose(2016)	61.8	84.9	67.5	57.1	68.2	66.5
AE(2017)	65.5	86.8	72.3	60.6	72.6	70.2
Top-down						
HG(2016)	63.0	85.7	68.9	58.0	70.4	68.0
CPN(2018)	72.1	91.4	80.0	68.7	77.2	78.5
SBN(2018)	73.7	91.9	81.1	70.3	80.0	79.0
HRNet(2019)	74.9	92.5	82.8	71.3	80.9	80.1
w/ AP-HNN						
Bottom-up						
OpenPose(2016)	63.3	85.7	68.8	58.5	69.4	68.1
AE(2017)	67.2	87.3	73.8	61.1	72.6	71.4
Top-down						
HG(2016)	65.1	86.8	70.9	59.9	72.1	70.0
CPN(2018)	74.0	91.6	81.6	70.3	78.5	79.8
SBN(2018)	75.1	91.7	82.5	71.5	81.3	80.1
HRNet(2019)	76.3	92.3	83.8	73.0	82.2	81.2

Table 6: Study of the AP-HNN generality on *test-dev2017*.

The predicted keypoints of AD-HNN are more accurate than those of SBN and HRNet with GCN in Fig. 7. It shows that the hypergraph-based modeling of the human pose is indeed effective and superior to the constant graph. For example, the self-occlusion cases lie in Fig. 7 while our method can better avoid this confusions, which shows that the global and local joint context can largely help infer the occluded joints.

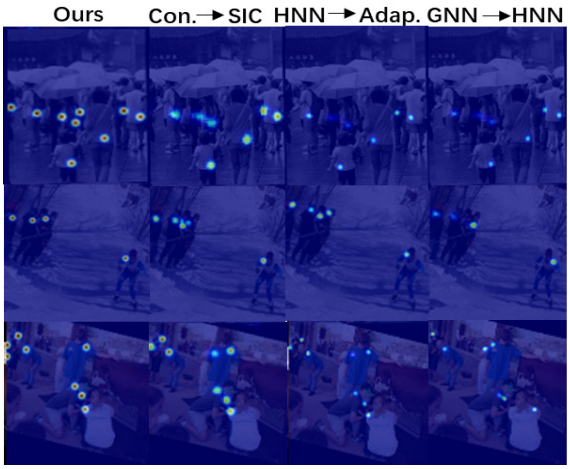

Figure 6: The visualized comparison of each module. "module A→ module B" means that replacing B with A.

High-order Semantic Learning Analysis. We further analyze the ability of AP-HNN to capture high-order joint information. Fig. 8 shows the predicted keypoints and the confidence score of GNN or HNN. We take the 'rshoulder' as example, its neighboring joint 'relbow' is accurately detected with a high confidence of 0.8. Similar confident results can be observed for GNN. This is possibly because both of them can well capture the neighboring structural information. However, if the 'lshoulder' is non-neighbored or 'rwrist' is far away from 'rshoulder', the predicted confidence score of GNN is obvious lower than HNN. The GNN-based model can't correctly recognize 'lshoulder' while HNN can. It shows that AP-HNN has superiority to capture high-order joint relationships.

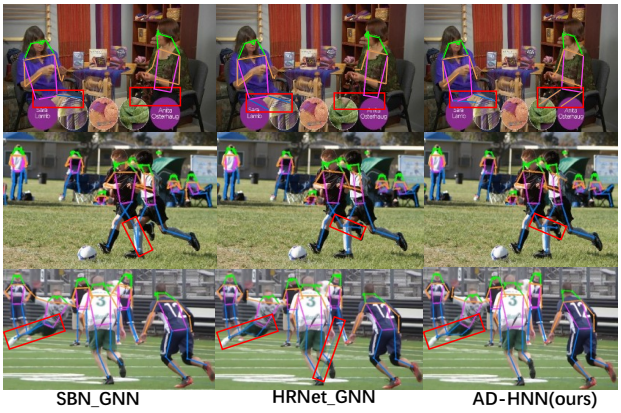

Figure 7: Qualitative comparison with the GNN-based baseline on self-occluded samples.

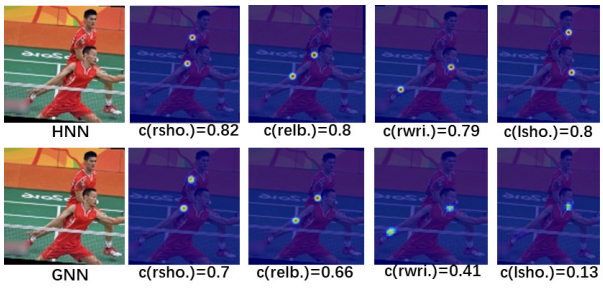

Figure 8: Comparison of the high-order semantic relations capturing capability, $c(\cdot)$ depicts the confidence score.

Conclusion

In this paper, we study the MPPE from a new perspective of introducing a novel ADaptive Hypergraph Neural Network (AD-HNN). To obtain better initial heatmap predictions, we design a semantic interaction convolution to explore the feature relation learning within the initial pose estimator. Build on this, we creatively propose an adaptive hypergraph to represent a human pose exploiting the feasible high-order semantic relations among joints flexibly. Unlike traditional Graph Convolutional Networks (GCNs) that adopt fixed tree structure, our AP-HNN can adaptively find the most reasonable structure for variable postures. Our method achieves almost best performance compared with the state-of-the-arts on both MS-COCO, MPII and CrowdPose datasets. In future, we plan to investigate the hypergraph learning in bottom-up MPPE.

References

Andriluka, M.; Pishchulin, L.; Gehler, P. V.; and Schiele, B. 2014. 2D Human Pose Estimation: New Benchmark and State of the Art Analysis. *CVPR*, 3686–3693.

Cao, Z.; Hidalgo, G.; Simon, T.; Wei, S.; and Sheikh, Y. 2019. OpenPose: Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1–1.

Chen, Y.; Shen, C.; Wei, X.; Liu, L.; and Yang, J. 2017. Adversarial PoseNet: A Structure-Aware Convolutional Network for Human Pose Estimation. *ICCV*, 1221–1230.

Chen, Y.; Wang, Z.; Peng, Y.; Zhang, Z.; Yu, G.; and Sun, J. 2018. Cascaded Pyramid Network for Multi-person Pose Estimation. *CVPR*, 7103–7112.

Cheng, B.; Xiao, B.; Wang, J.; Shi, H.; Huang, T. S.; and Zhang, L. 2020. HigherHRNet: Scale-Aware Representation Learning for Bottom-Up Human Pose Estimation. *CVPR*, 5386–5395.

Chou, C.; Chien, J.; and Chen, H. 2018. Self Adversarial Training for Human Pose Estimation. *APSIPA ASC*, 17–30.

Chu, X.; Yang, W.; Ouyang, W.; Ma, C.; Yuille, A. L.; and Wang, X. 2017. Multi-context Attention for Human Pose Estimation. *CVPR*, 5669–5678.

Fang, H.; Xie, S.; Tai, Y.; and Lu, C. 2017. RMPE: Regional Multi-person Pose Estimation. *In Proceedings of the IEEE International Conference on Computer Vision*, 2353–2362.

Feng, Y.; You, H.; Zhang, Z.; Ji, R.; and Gao, Y. 2019. Hypergraph Neural Networks. *IJCAI*, 33(01): 3558–3565.

Huang, J.; Zhu, Z.; Guo, F.; and Huang, G. 2020. The Devil Is in the Details: Delving Into Unbiased Data Processing for Human Pose Estimation. *CVPR*, 5699–5708.

Insafutdinov, E.; Pishchulin, L.; Andres, B.; Andriluka, M.; and Schiele, B. 2016. DeeperCut: A Deeper, Stronger, and Faster Multi-person Pose Estimation Model. *ECCV*, 34–50.

Jiang, J.; Wei, Y.; Feng, Y.; Cao, J.; and Gao, Y. 2019. Dynamic Hypergraph Neural Networks. *IJCAI*, 2635–2641.

Jin, S.; Liu, W.; Xie, E.; Wang, W.; and Luo, P. 2020. Differentiable Hierarchical Graph Grouping for Multi-Person Pose Estimation. *ECCV*, 718–734.

Ke, L.; Chang, M.; Qi, H.; and Lyu, S. 2018. Multi-Scale Structure-Aware Network for Human Pose Estimation. *ECCV*, 731–746.

Kim, E.; Kang, W. Y.; On, K.-W.; Heo, Y.-J.; and Zhang, B. 2020. Hypergraph Attention Networks for Multimodal Learning. *CVPR*, 4569–4578.

Kingma, D. P.; and Ba, J. 2015. Adam: A Method for Stochastic Optimization. *ICLR*.

Kipf, T. N.; and Welling, M. 2016. Semi-Supervised Classification with Graph Convolutional Networks. *arXiv preprint arXiv:1609.02907*.

Li, J.; Wang, C.; Zhu, H.; Mao, Y.; Fang, H.; and Lu, C. 2019. CrowdPose: Efficient Crowded Scenes Pose Estimation and A New Benchmark. *CVPR*, 10863–10872.

Li, Y.; Li, K.; Jiang, S.; Zhang, Z.; Huang, C.; and Da Xu, R. Y. 2020. Geometry-driven self-supervised method for 3D human pose estimation. 34(07): 11442–11449.

Lin, T.; Maire, M.; Belongie, S. J.; Hays, J.; Perona, P.; Ramanan, D.; Dollar, P.; and Zitnick, C. L. 2014. Microsoft COCO: Common Objects in Context. *ECCV*, 740–755.

Liu, S.; Lv, P.; Zhang, Y.; Fu, J.; Cheng, J.; Li, W.; Zhou, B.; and Xu, M. 2020. Semi-Dynamic Hypergraph Neural Network for 3D Pose Estimation. *IJCAI*, 782–788.

Luvizon, D. C.; and Picard, D. 2019. Human Pose Regression by Combining Indirect Part Detection and Contextual Information. *Computers & Graphics*, 85: 15–22.

Moon, G.; Chang, J. Y.; and Lee, K. M. 2019. PoseFix: Model-Agnostic General Human Pose Refinement Network. *CVPR*, 7765–7773.

Newell, A.; and Deng, J. 2017. Associative Embedding: End-to-End Learning for Joint Detection and Grouping. *NIPS*, 2277–2287.

Newell, Y.; and Kaiyu, J., Deng. 2016. Stacked Hourglass Networks for Human Pose Estimation. *ECCV*, 483–499.

Ning, G.; Zhang, Z.; and He, Z. 2018. Knowledge-Guided Deep Fractal Neural Networks for Human Pose Estimation. *IEEE Transactions on Multimedia*, 1246–1259.

Papandreou, G.; Zhu, T.; Kanazawa, N.; Toshev, A.; Tompson, J.; Bregler, C.; and Murphy, K. P. 2017. Towards Accurate Multi-person Pose Estimation in the Wild. *CVPR*, 3711–3719.

Qiu, L.; Zhang, X.; Li, Y.; Li, G.; Wu, X.; Xiong, Z.; Han, X.; and Cui, S. 2020a. Peeking into occluded joints: A novel framework for crowd pose estimation. *ECCV*, 488–504.

Qiu, Z.; Qiu, K.; Fu, J.; and Fu, D. 2020b. Dgcn: Dynamic graph convolutional network for efficient multi-person pose estimation. 34(07): 11924–11931.

Russakovsky, O.; Deng, J.; Su, H.; Krause, J.; Satheesh, S.; Ma, S.; Huang, Z.; Karpathy, A.; Khosla, A.; Bernstein, M. S.; et al. 2015. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision*, 115(3): 211–252.

Su, K.; Yu, D.; Xu, Z.; Geng, X.; and Wang, C. 2019. Multi-Person Pose Estimation with Enhanced Channel-wise and Spatial Information. *CVPR*, 5674–5682.

Sun, K.; Lan, C.; Xing, J.; Zeng, W.; Liu, D.; and Wang, J. 2017. Human Pose Estimation Using Global and Local Normalization. *ICCV*, 5600–5608.

Sun, K.; Xiao, B.; Liu, D.; and Wang, J. 2019. Deep High-Resolution Representation Learning for Human Pose Estimation. *CVPR*, 5693–5703.

Tang, W.; Yu, P.; and Wu, Y. 2018. Deeply Learned Compositional Models for Human Pose Estimation. *ECCV*, 197–214.

Wang, J.; Long, X.; Gao, Y.; Ding, E.; and Wen, S. 2020. Graph-PCNN: Two Stage Human Pose Estimation with Graph Pose Refinement. *ECCV*, 492–508.

Wei, S.; Ramakrishna, V.; Kanade, T.; and Sheikh, Y. 2016. Convolutional Pose Machines. *CVPR*, 4724–4732.

Xiao, B.; and Wei, Y. 2018. Simple Baselines for Human Pose Estimation and Tracking. *ECCV*, 472–487.

Yan, S.; Xiong, Y.; Lin, D.; and Tang, X. 2018. Spatial Temporal Graph Convolutional Networks for Skeleton-Based Action Recognition. *AAAI*, 7444–7452.

Yan, Y.; Qin, J.; Chen, J.; Liu, L.; Zhu, F.; Tai, Y.; and Shao, L. 2020. Learning Multi-Granular Hypergraphs for Video-Based Person Re-Identification. *CVPR*, 2896–2905.

Yang, W.; Li, S.; Ouyang, W.; Li, H.; and Wang, X. 2017. Learning Feature Pyramids for Human Pose Estimation. *ECCV*, 1290–1299.

Yu, J.; and Tao, D. 2012. Adaptive Hypergraph Learning and its Application in Image Classification. *IEEE Transactions on Image Processing*, 21(7): 3262–3272.

Zhang, F.; Zhu, X.; Dai, H.; Ye, M.; and Zhu, C. 2020. Distribution-Aware Coordinate Representation for Human Pose Estimation. *CVPR*, 7091–7100.

Zhu, X.; Zhu, Y.; Zhang, S.; Hu, R.; and He, W. 2017. Adaptive Hypergraph Learning for Unsupervised Feature Selection. *IJCAI*, 3581–3587.